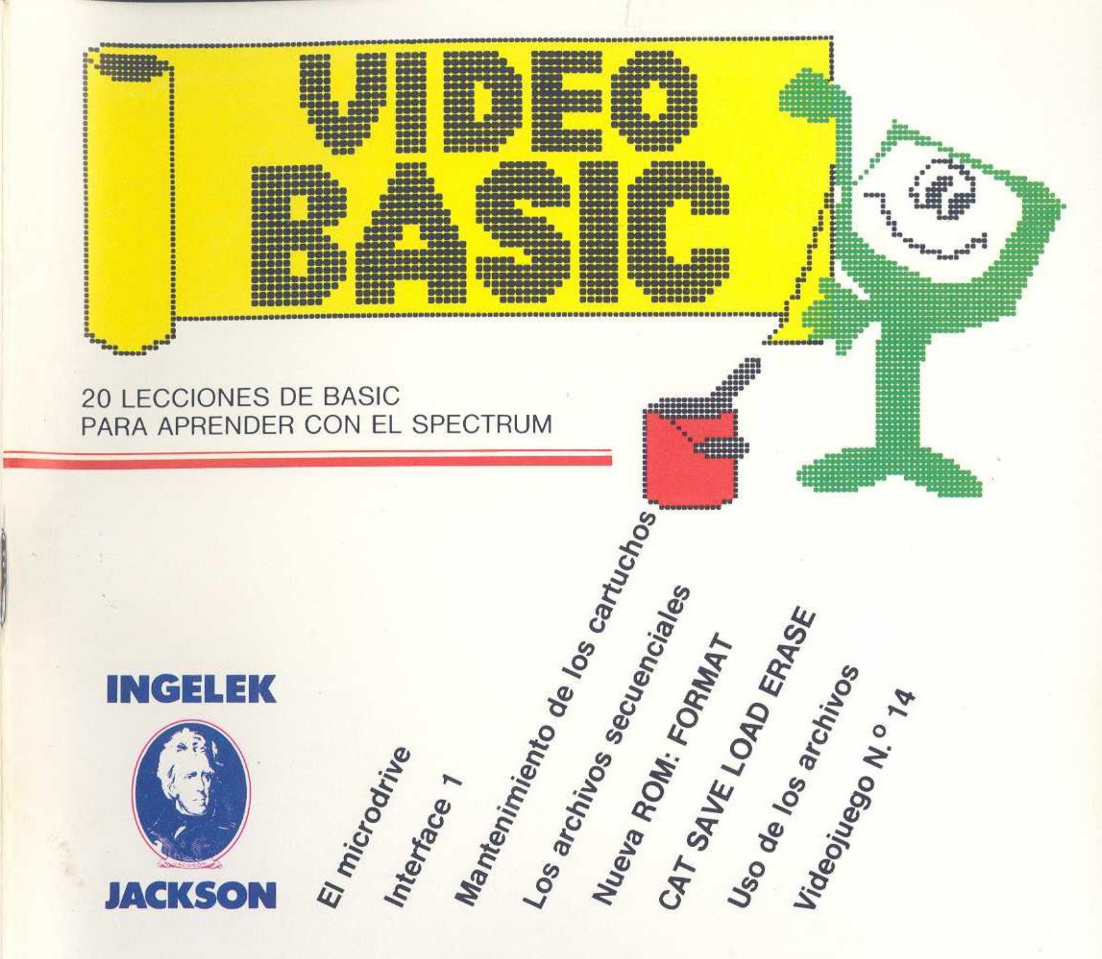


20 LECCIONES DE BASIC
PARA APRENDER CON EL SPECTRUM

INGELEK

JACKSON

El microdrive

Interface 1

Mantenimiento de los cartuchos

Los archivos secuenciales

Nueva ROM: FORMAT

CAT SAVE LOAD ERASE

Uso de los archivos

Videojuego N.º 14

14

Spectrum

16K/48K/PLUS

VIDEO BASIC
Una publicación de
INGELEK JACKSON
Director editor por INGELEK:
Antonio M. Ferrer
Director editor por JACKSON HISPANIA:
Lorenzo Bertagnolio
Director de producción:
Vicente Robles
Autor: Softidea
Redacción software italiano:
Francesco Franceschini,
Stefano Cremonesi
Redacción software castellano:
Fernando López, Antonio Carvajal,
Alberto Caffarato, Pilar Manzanera
Diseño gráfico:
Studio Nuovaidea
Ilustraciones:
Cinzia Ferrari, Silvano Scolari,
Equipo Galata
Ediciones INGELEK, S. A.
Dirección, redacción y administración,
números atrasados y suscripciones:
Avda. Alfonso XIII, 141
28016 Madrid. Tel. 2505820
Fotocomposición: Espacio y Punto, S. A.
Imprime: Gráficas Reunidas, S. A.

ISBN del tomo 4: 84-85831-20-9
ISBN del fascículo: 84-85831-11-X
ISBN de la obra completa: 84-85831-10-1
Depósito Legal: M-15076-1985
Plan general de la obra:
20 fascículos y 20 casetes, de aparición quincenal, coleccionables en 5 estuches.
Distribución en España:
COEDIS, S. A.
Valencia, 245. 08007 Barcelona.
INGELEK JACKSON garantiza la publicación de todos los fascículos y casetes que componen esta obra y el suministro de cualquier número atrasado o estuche mientras dure la publicación y hasta un año después de terminada.
El editor se reserva el derecho de modificar el precio de venta del fascículo, en el transcurso de la obra, si las circunstancias del mercado así lo exigen.
Octubre, 1985.
Impreso en España.

INGELEK



JACKSON

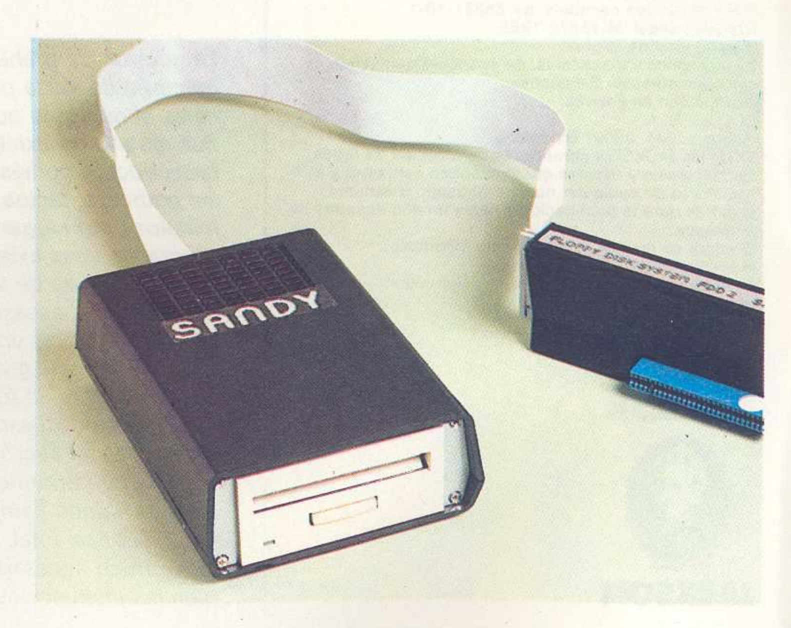
SUMARIO

Introducción

La económica grabadora constituye la memoria de masa por excelencia, hasta el punto de que se puede decir que sin ella no existirían los ordenadores domésticos. Aburren, sin embargo, tantos interminables minutos para cargar o guardar un programa, esas extenuantes esperas en la búsqueda de un programa concreto, y la escasa fiabilidad que poseen todas las operaciones en las que interviene la grabadora.
Como siempre, la tecnología ha sabido dar respuesta a todos estos problemas: el Microdrive.
Tratar ahora enormes masas de datos, en poco tiempo y con una fiabilidad casi total, resulta fácil; el único truco necesario es conocer bien las instrucciones.

El Microdrive

En lecciones anteriores hemos visto ya cómo es posible –a través de una grabadora de casete– memorizar en cinta magnética todos los programas y los datos que de otra manera hubiéramos perdido irremisiblemente al apagar el ordenador.
La grabación y la lectura de cinta –seguramente habrás podido darte cuenta ya– es una operación muy sencilla, fiable y económica; sin embargo, necesita largas esperas para la carga de las informaciones y, sobre todo, precisa de una intervención manual para poner en marcha y apagar el motor de la grabadora (el famoso "START TAPE").
La búsqueda de los programas sobre la cinta está garantizada por una técnica bastante empírica, que consiste en contar los segundos, o apuntar el número del contador de cinta del aparato, con el riesgo permanente (sobre todo cuando uno es bastante distraido) de grabar sobre una zona ya grabada anteriormente provocando la pérdida de un programa que



quizá nos haya costado largas horas de paciente teclear.
Una de las posibilidades más interesantes para ampliar las prestaciones ofrecidas por tu Spectrum consiste seguramente en la unidad de Microdrive.
El objeto del Microdrive, del cual nos ocuparemos en esta lección, es precisamente la eliminación de todos los «inconvenientes» antes citados, permitiendo una grabación tan segura como la ofrecida por la grabadora, pero mucho más rápida y, sobre todo, completamente automática.
Cuando se utiliza un Microdrive no es necesario, como ocurriera con la grabadora, ocuparse para nada de su funcionamiento: la gestión de las diversas operaciones —como por ejemplo la grabación o la carga de los programas— tiene lugar en este caso bajo el control directo de la unidad central, sin ninguna intervención exterior de búsqueda o verificación.

Cómo funciona un Microdrive

Un Microdrive consiste asencialmente en un dispositivo, cuyo funcionamiento resulta muy parecido al de una grabadora de casete, capaz de leer o escribir —mediante un cabezal incluido en su interior— la cinta magnética contenida en los cartuchos (o cartridge), que se introduce en el propio lector de Microdrive.
El cartucho consiste en un estuche de plástico en cuyo interior se encuentra una cinta magnética enrrollada como un sinfin, es decir, de manera que no necesita ningún rebobinado para la lectura o la escritura de las informaciones.
El avance de la cinta tiene lugar siempre en la misma dirección, permitiendo así al sistema alcanzar una velocidad de lectura y escritura de datos de otra forma inalcanzable.
Esta solución comporta, sin embargo, algunos inconvenientes, especialmente en lo relacionado con el tiempo de acceso a las informaciones (entendiendo como «tiempo de acceso» aquél necesario para la búsqueda y carga de los datos o los programas grabados).
Si por ejemplo, deseamos cargar un programa cuyo comienzo, por azar, se encuentra justo debajo del cabezal, es

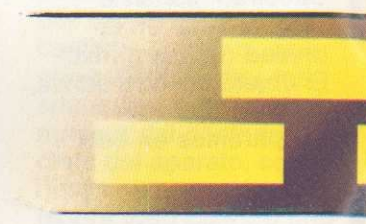
necesario hacer correr toda la cinta (5 metros) antes de que sea posible acceder a él, (puesto que no es posible rebobinarla) empleando así demasiado tiempo para hacer una sola búsqueda. De todas maneras, ten siempre presente –para tener una idea más clara de lo que significa «tiempo»– que la cinta, gracias a la gran velocidad del motor, emplea aproximadamente 7 segundos para completar un ciclo completo. ¡Nada que ver con un normal casete de audio!

Es interesante deternerse un momento para explicar como se graban los datos en la cinta.

En el Microdrive, al contrario de lo que ocurre con una grabadora, las informaciones son memorizadas según bloques de datos de dimensiones fijas llamados "sectores". Puedes imaginarte los sectores como distintas zonas de la cinta, en cada una de la cuales es posible escribir los datos.

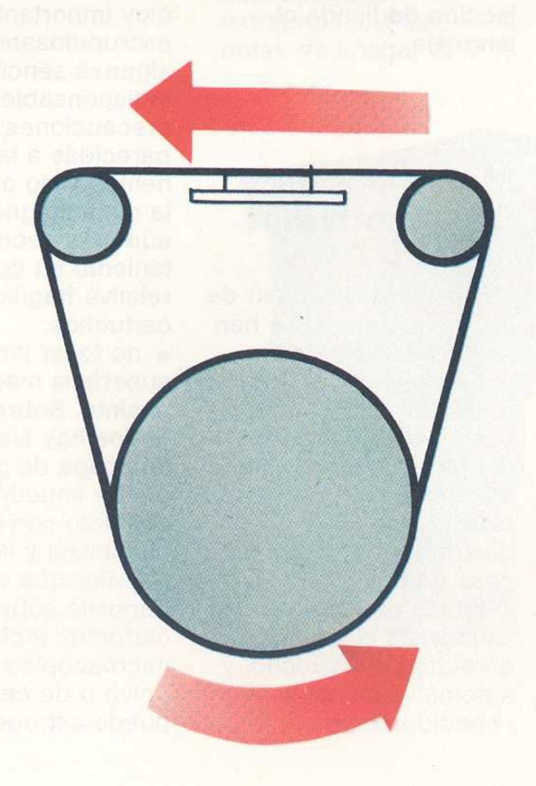
La primera vez que la cinta se pone en marcha, estos sectores, naturalmente, no existen: es necesario entonces ejecutar una operación especial –llamada "formateado"– que proporciona al Microdrive un sistema de referencias que está constituido precisamente por estos sectores.

Podríamos decir, para que lo entiendas claramente, que en el momento del formateado, la cinta se



despedaza en muchos segmentos, cada uno de los cuales queda preparado para grabar una determinada cantidad de informaciones. Trataremos ahora de comprender cuál puede ser la utilidad de esta maniobra. Inmediatamente después del formateado, todos los sectores quedan libres; las sucesivas instrucciones de grabación irán llenando paulatinamente los sectores uno tras otro, mientras que las instrucciones de cancelación los podrán vaciar.

Disposición de los datos sobre la cinta del microdrive.



El sinfin de cinta contenido en un cartucho de microdrive tiene 5 metros de longitud.

La gran ventaja de haber formateado la cinta es el hecho de que un mismo programa no tendrá porqué ser grabado necesariamente en sectores contiguos, permitiéndote así disfrutar de todos los sectores que los borrados han ido liberando en todos los puntos de la cinta.

En otras palabras, el formateado permite aprovechar al máximo los recursos que ofrece la cinta magnética (eliminando así cualquier pérdida de espacio y tiempo) facilitando además a los circuitos de control del Microdrive comprender fácilmente sobre que punto de la cinta se encuentra el cabezal de lectura/escritura. Profundizaremos de todas maneras en el tema del formateado en la parte de nuestra lección dedicada al lenguaje.

Mantenimiento de los cartuchos

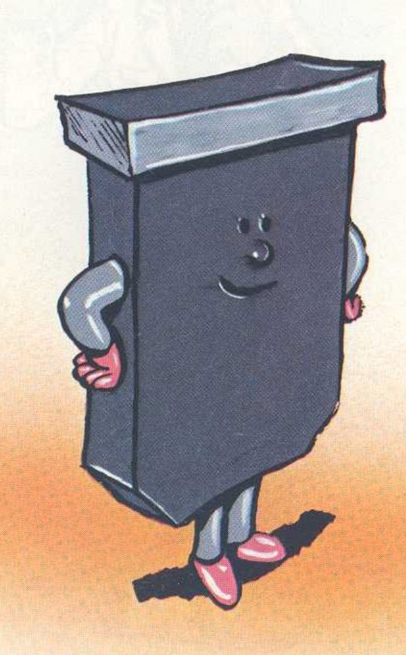
Los casetes disponen de lengüetas que, si se han arrancado, impiden cualquier borrado o grabación accidental. También los cartuchos del Microdrive permiten esto: es suficiente con romper con un destornillador u otra cosa parecida la lengüeta de plástico situada en el lado derecho del cartucho, y automáticamente será impedido cualquier intento de escritura de nuevas informaciones en la cinta de los cartuchos.
Si tras algún tiempo se desea reutilizar este cartucho es suficiente con tapar el espacio de la lengüeta con un poco de cinta adhesiva.
Hemos visto ya que, a pesar de sus reducidas dimensiones, un minúsculo cartucho es capaz de grabar con precisión una enorme cantidad de informaciones; esta es la razón de que sea muy importante tomar escrupulosamente algunas sencillas, pero indispensables, precauciones, parecidas a las que ya hemos visto al tratar de la cinta magnética, pero aún más necesarias, teniento en cuenta la relativa fragilidad de los cartuchos:

- no tocar jamás la superficie magnética de la cinta. Sobre los dedos hay siempre una fina capa de grasa que puede impedir el contacto correcto entre el cabezal y la cinta.
- evitar que el polvo se deposite sobre el cartucho: incluso un microscópico grano de polvo o de ceniza puede estroperar

irremisiblemente la capa magnética. La mejor solución es devolver cada cartucho a su estuche protector inmediatamente después de usarlo;

- no exponer los cartuchos al sol, al calor o a las bajas temperaturas, puesto que podrían pruducir peligrosas deformaciones del soporte plástico que impediría el buen arrastre de la cinta;
- alejar todo lo posible los cartuchos de los imanes, televisores, altavoces y, en general, cualquier aparato eléctrico;
- hacer una copia de seguridad de todos los cartuchos que contengan informaciones importantes, conservándolos además en un lugar distinto de aquél en que están los originales;
- no dejar nunca los cartuchos en el lector cuando el ordenador está apagado: extraerlos siempre antes de apagar la máquina y guardarlos inmediatamente.

Antes de concluir nuestro capítulo es necesario recordar que conviene sustituir rápidamente los cartuchos que, después de un uso prolongado o una caída, presenten algunos fallos de cierta importancia (como pequeñas paradas o bloqueos durante la rotación del cartucho): tu Microdrive te estará siempre agradecido, pagándote con un trabajo siempre rápido, y, sobre todo, fiable.



Los canales

Tu Spectrum posee, para dialogar con los diversos periféricos de entrada y salida, una estructura muy sofisticada, basada principalmente en los flujos de información y los canales.
Esta estructura se usa muy raramente (salvo en casos especiales) en los Spectrum de configuración estándar (es decir, que carecen de interface 1), ya que son pocos los dispositivos privados de instrucciones expresamente creadas para su uso. Por ejemplo, el envío de las informaciones a pantalla puede ejecutarse con la instrucción PRINT, mientras que para enviar cualquier cosa a impresora se puede



utilizar la instrucción LPRINT.
Pero la adición de un Microdrive requiere que profundicemos más aún en este tema, ya que la mucha flexibilidad que nos ofrece este periférico pone de manifiesto todas las ventajas de una estructura como ésta. Una forma bastante sencilla de comprender el significado y la diferencia entre flujo y canal es considerar una unidad periférica cualquiera, como si estuviera subdividida en dos partes separadas. La primera, físicamente visible, es la que envía o recibe las informaciones desde la unidad central; se llama canal. Ejemplos de canales hacia los cuales el Spectrum puede enviar datos son la pantalla o la impresora; los canales desde los cuales la unidad central puede recibir datos son en cambio el teclado o la grabadora.
La segunda parte, sin embargo, tiene relación con el recorrido que los datos deben seguir para llegar o salir de los canales. Estos recorridos se llaman flujos. En el Spectrum existen 16 flujos disponibles, cuatro de los cuales quedan automáticamente reservados, en el momento del encendido, para determinados canales. Los flujos 0 y 1 son idénticos y están conectados al canal "K". Este canal envía

NUMERO DE FLUJO UTILIZACION DE ENTRADA	IDENTIFICACION DEL CANAL	UTILIZACION DE SALIDA
# 0 teclado	"K"	Pantalla inf.
# 1 teclado	"K"	Pantalla inf.
# 2 ///	"S"	Pantalla sup.
# 3 ///	"P"	Impresora

la salida de las informaciones a la parte baja de la pantalla (es decir, donde aparecen los mensajes de error y los INPUT) y recibe como entrada los datos desde el teclado. El flujo 2 está, en cambio, comunicado con el canal "S", y visualiza los caracteres en la parte superior de la pantalla; el flujo 3 está conectado al canal "P" y envía las informaciones a la impresora.
No es posible utilizar los flujos 2 y 3 para entrada: cualquier intento de leer algo, por ejemplo del flujo 3, producirá el mensaje de error.

```
INVALID I/O DEVICE
```

El símbolo del flujo es el (#).
Por el momento nos referimos al Spectrum todavía no conectado al Microdrive y al interface 1.

OPEN/CLOSE

El concepto de flujo es bastante fácil de comprender; puede ocurrir, sin embargo, que tú te estés preguntando qué hacer, y qué utilidad tiene asociar los números de flujo con los dispositivos de entrada/salida. Encontrarás la respuesta en seguida. Antes de nada diremos que la unidad central, antes de poder dialogar con un periférico a través de un flujo, debe abrir (OPEN) este flujo. La apertura del flujo tiene un doble objeto. En primer lugar asocia un número de flujo a un determinado dispositivo de E/S; en segundo lugar, señala el dispositivo de E/S que deberá ser utilizado. Para abrir un flujo, el BASIC del Spectrum utiliza la instrucción

```
OPEN # F, C
```

en la que "F" es el número de flujo a abrir, mientras que "C" es la cadena que especifica el canal al que ha de asociarse ese flujo. Una vez enviada esta instrucción cualquier dato enviado o leído a través del flujo recién abierto se referirá al canal "C".
Así

```
OPEN # 5, "K"
```

abrirá el flujo 5, asociándolo al teclado. Ahora, si mandamos esta instrucción:

```
INPUT # 5; A, B
```

podremos asignar valores a las variables A y B igual que a través de una normal instrucción INPUT, ya

que el flujo se ha abierto hacia el teclado (KEYBOARD).
Si, en cambio, ejecutamos:

```
PRINT # 5; "ADIOS A TODOS"
```

enviaremos la cadena "ADIOS A TODOS" al flujo 5, que corresponde al área inferior de la pantalla. Conseguiremos así escribir con una instrucción PRINT en la parte de la pantalla normalmente reservada a los mensajes de INPUT.
Desgraciadamente, ejecutando esta instrucción la cadena no es visible (la parte inferior de la pantalla se "limpia" en el momento en que se detiene el programa).
Para ver los efectos de este flujo intenta ahora ejecutar este breve programa:

```
10 OPEN # 5, "K"
20 PRINT # 5; RND
30 GOTO 20
```

¡Esta vez en cambio el efecto será bien visible! El programa, sin embargo, finalizará con un mensaje de error, ya que el área de entrada no realiza el "scroll" de la misma forma que en el área de impresión normal.
En principio, cualquier flujo puede funcionar tanto en entrada como en salida; en la práctica, el único canal capaz de aceptar datos tanto en entrada como en salida es el que acabamos de ver en el ejemplo, es decir, el canal del teclado. Los otros dos –pantalla e impresora– son canales de sólo salida, y cualquier intento de leer alguna cosa en ellos provocará un mensaje de error.
Esta limitación se debe exclusivamente al hardware conectado al flujo; en otras palabras, es independiente de este último.
Para desconectar un flujo de un canal existe otra instrucción, CLOSE, cuyo objeto es precisamente separar lo que OPEN había unido.
En nuestro anterior ejemplo deberíamos haber escrito:

```
CLOSE # 5
```

Así, antes habríamos abierto el flujo número cinco. Ahora lo habríamos cerrado. CLOSE efectúa una operación que

aparentemente parece inútil, pero que, por el contrario, resulta importantísima para el buen funcionamiento del ordenador. El cierre de los flujos permite evitar la espera de datos del dispositivo conectado al flujo, evitando posibles errores y malos funcionamientos.

Cómo se usan los flujos

La única ventaja que hemos podido apreciar hasta ahora del uso de los flujos es la de conseguir escribir en la parte inferior de la pantalla. Pero existen además una serie de óptimas razones para usar los flujos, y la primera y más importante de todas ellas se llama "independencia de los periféricos". Este término, aparentemente carente de sentido, se ilustra fácilmente con un ejemplo.
Supón que queramos escribir un programa que produzca una serie de números al azar en pantalla, o bien los imprima con la impresora.
Utilizando PRINT y LPRINT, el programa resultaría una cosa muy parecida a ésta:

```
10 INPUT "IMPRESORA O PANTALLA P/S"; A$
20 IF A$ (1) = "P" THEN LPRINT RND
30 IF A$ (1) = "S" THEN PRINT RND
40 GOTO 20
```

Con el flujo, en cambio, podríamos escribir:

```
10 INPUT "IMPRESORA O PANTALLA P/S"; A$
20 OPEN # 5, A$ (1)
30 PRINT # 5; RND
40 GOTO 30
```

Pero aunque este ejemplo es demasiado corto para resultar eficazmente convincente, puedes estar seguro de que en programas largos el uso de los flujos es extremadamente práctico y ventajoso. Usando el Microdrive los flujos resultan además inevitables: esperamos que esta introducción a los flujos te haya animado a su uso.

Los archivos

Los programas son bastante útiles pero, en verdad, aquello que realmente se desea de un ordenador es el tratamiento de informaciones, datos, números, nombres, direcciones y cantidades.
Cualquier instrucción que pueda ser escrita puede también ser introducida en la memoria del ordenador, y, en consecuencia, memorizada en un cartucho.
Todos tenemos una cierta experiencia de archivo, ya sea del tratamiento de archivos propio de una oficina,

del archivo de estados de la cuenta corriente del banco, o, incluso, de las recetas de los platos preferidos.
Sabemos perfectamente la importancia que tiene archivar ordenadamente nuestras cosas.
Un archivo (en inglés, "file") es un conjunto de informaciones que, por un motivo cualquiera, se desea mantener agrupadas.
Ejemplos de archivos pueden ser los nombres de los alumnos de una clase, los números extraidos durante una partida de bingo, los nombres y las direcciones de aquellos nacidos en un día determinado, o un programa.
El Microdrive permite a tu Spectrum la creación de archivos del tipo denominado secuencial. Esta denominación deriva del hecho de que los elementos que componen el archivo son grabados uno detrás de otro, esto es, en orden secuencial.
La razón de esta solución es fácilmente explicable por la propia constitución física del



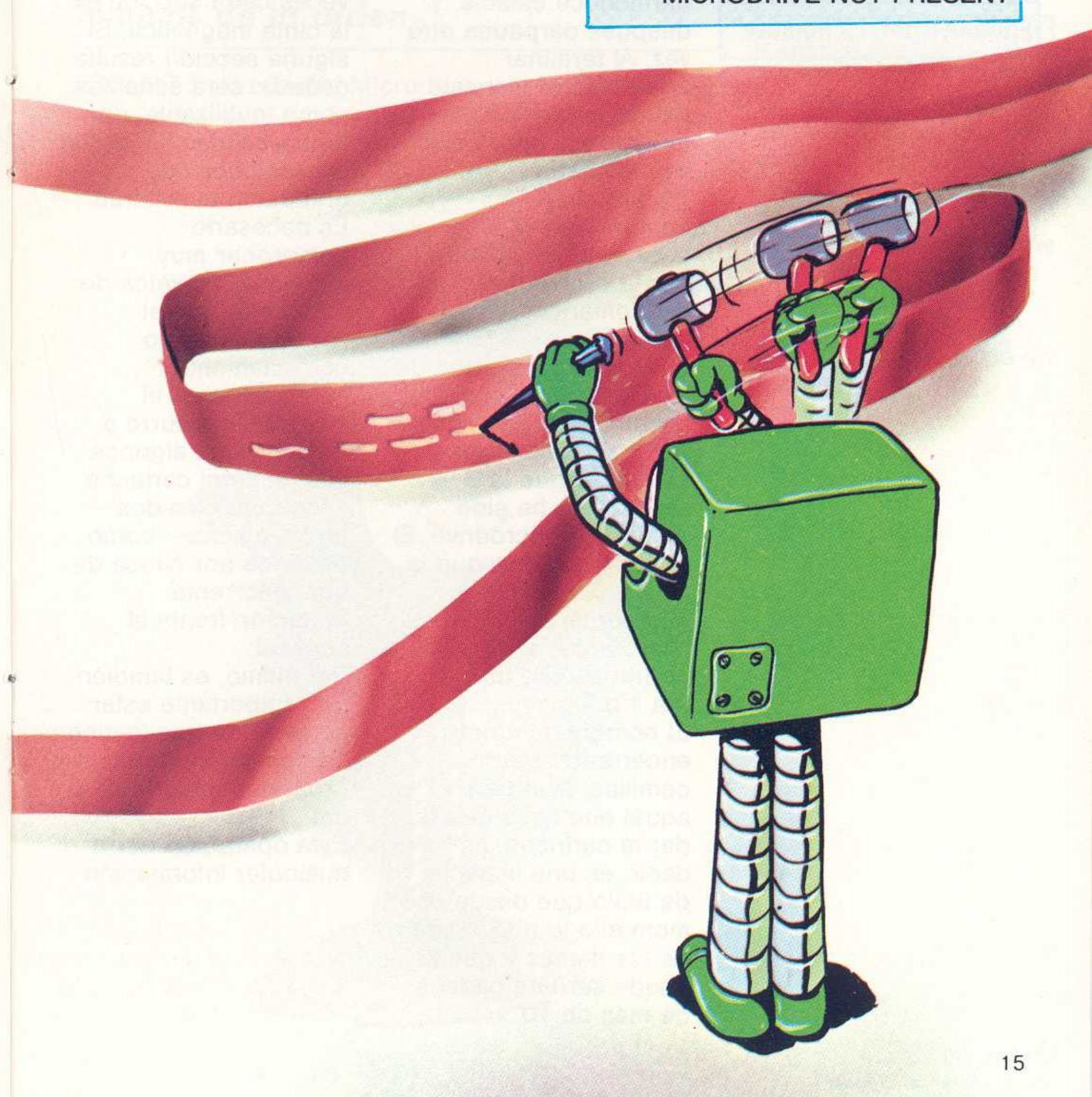
soporte utilizado por el Microdrive: la larga cinta magnética guardada dentro del cartucho constituye de hecho una estructura ideal para guardar informaciones según un orden de este tipo.
Examinemos, por tanto, qué debemos hacer para guardar en los cartuchos, reuniéndolos, todos los datos e informaciones que desearíamos que no se perdieran para siempre al apagar el ordenador.
Para hacer esto es necesario desde el principio introducir dos términos nuevos, muy útiles cuando se habla de archivos: Registros y Campos.
Hagámoslo (para ser lo más claros posible) a través de un ejemplo.
La guía telefónica de tu ciudad, considerada en conjunto, es un archivo: contiene informaciones ordenadas y clasificadas según un orden bien definido.
Este archivo está constituido por millares de registros, cada uno de los cuales está compuesto por un apellido, un nombre, una dirección y un número de teléfono.
Así, un registro puede ser definido como un grupo de datos relacionados entre sí; cada uno de los datos concretos que introducimos en el registro se llaman, en cambio, campos.
En nuestro caso, existen 4 campos para cada registro: un campo Apellido, un campo Nombre, un campo Dirección y un campo Número de teléfono.
La elección del formato del registro (es decir, del número de campos) es, por supuesto, arbitraria: habriamos podido considerar, junto con los campos ya citados, también un campo Número de prefijo, que indicará el prefijo telefónico.
Hemos dicho ya que en los archivos secuenciales, como indica su propio nombre, las informaciones son archivadas una detrás de otra. Esto significa que para leer (o escribir) la última información del archivo en uso es necesario haber leído (o escrito) antes todas las informaciones anteriores.
Esta limitación puede parecer una notable desventaja (también sería más cómodo para nosotros encontrarnos, cuando lo estuvieramos buscando, con “López, Juan” nada más abrir la guía telefónica) pero lo cierto es que –gracias a la notable velocidad de lectura de los datos en el Microdrive– encontrar un registro aislado contenido en un archivo de dimensiones medias requiere un tiempo de búsqueda bastante razonable.
Comprobaremos y profundizaremos en este tema en la parte de la lección dedicada a la programación.

FORMAT

La primera cosa que hay que hacer con un cartucho virgen (es decir, nunca usado) es formatearlo.

Cualquier intento de utilizarlo antes de haber ejecutado esta operación provocará la aparición del mensaje de error:

MICRODRIVE NOT PRESENT



En otras palabras, el lector no reconocerá el cartucho.
Para poderlo formatear es necesario, antes de nada, insertarlo en el Microdrive, y después escribir:

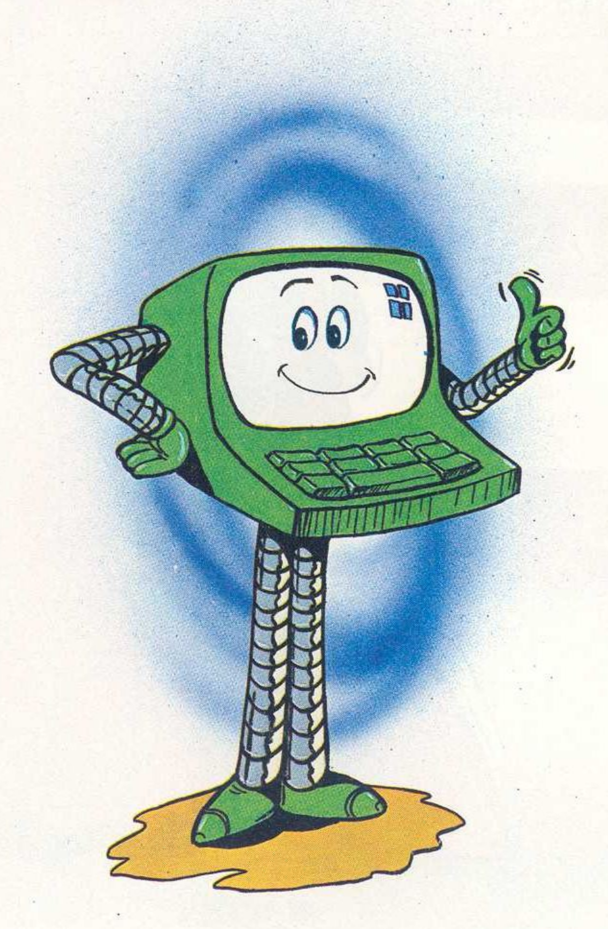
```
FORMAT "m";1; "nombre"
```

Apenas hayas pulsado la tecla ENTER notarás que el motor se pone en marcha y, por un período de unos 30 segundos, verás que la pantalla primero parpadea, luego permanece estable, y, después parpadea otra vez. Al terminar aparecerá el mensaje OK: la operación de formateado habrá sido ya ejecutada.
Tratemos de comprender qué ha sucedido comenzando por la propia sintaxis del comando. La instrucción FORMAT es, obviamente, la orden de formateado de la cinta, mientras que la "m" (fíjate en las comillas) establece que la instrucción ha sido referida al Microdrive. El número 1 indica que la cinta está en el Microdrive 1; con otros Microdrives esta cifra podría asumir otro valor (de 1 a 8).
El nombre, también encerrado entre comillas, se refiere a aquél que se le desea dar al cartucho, es decir, es una especie de título que desde ese momento lo distinguirá de los demás y que no puede ser una cadena de más de 10 caracteres.
Durante los 30 segundos, el ordenador "configurará" el cartucho, memorizando el nombre especificado junto con FORMAT y comprobando varias veces cada sección de la cinta magnética. Si alguna sección resulta dañada, será señalada como inutilizable, de forma que las sucesivas operaciones de grabación la evitarán.
Es necesario comprobar muy atentamente, antes de formatear, que el cartucho ha sido correctamente introducido en el Microdrive: ocurre a menudo que algunos sectores del cartucho sean considerados –erróneamente– como dañados por causa de una incorrecta inserción frente al cabezal.
Por último, es también muy importante estar seguros de que la cinta no contiene ningún programa importante o útil.
Esta operación borra cualquier información

contenida anteriormente en el cartucho, eliminándola definitivamente.
Por esto es necesario prestar atención al contenido de la cinta antes de formatearla, ya que, muchas veces, se recurre a esta orden para dejar el cartucho "como nuevo".

Sintaxis de la orden

FORMAT "m"; número Microdrive; "nombre"



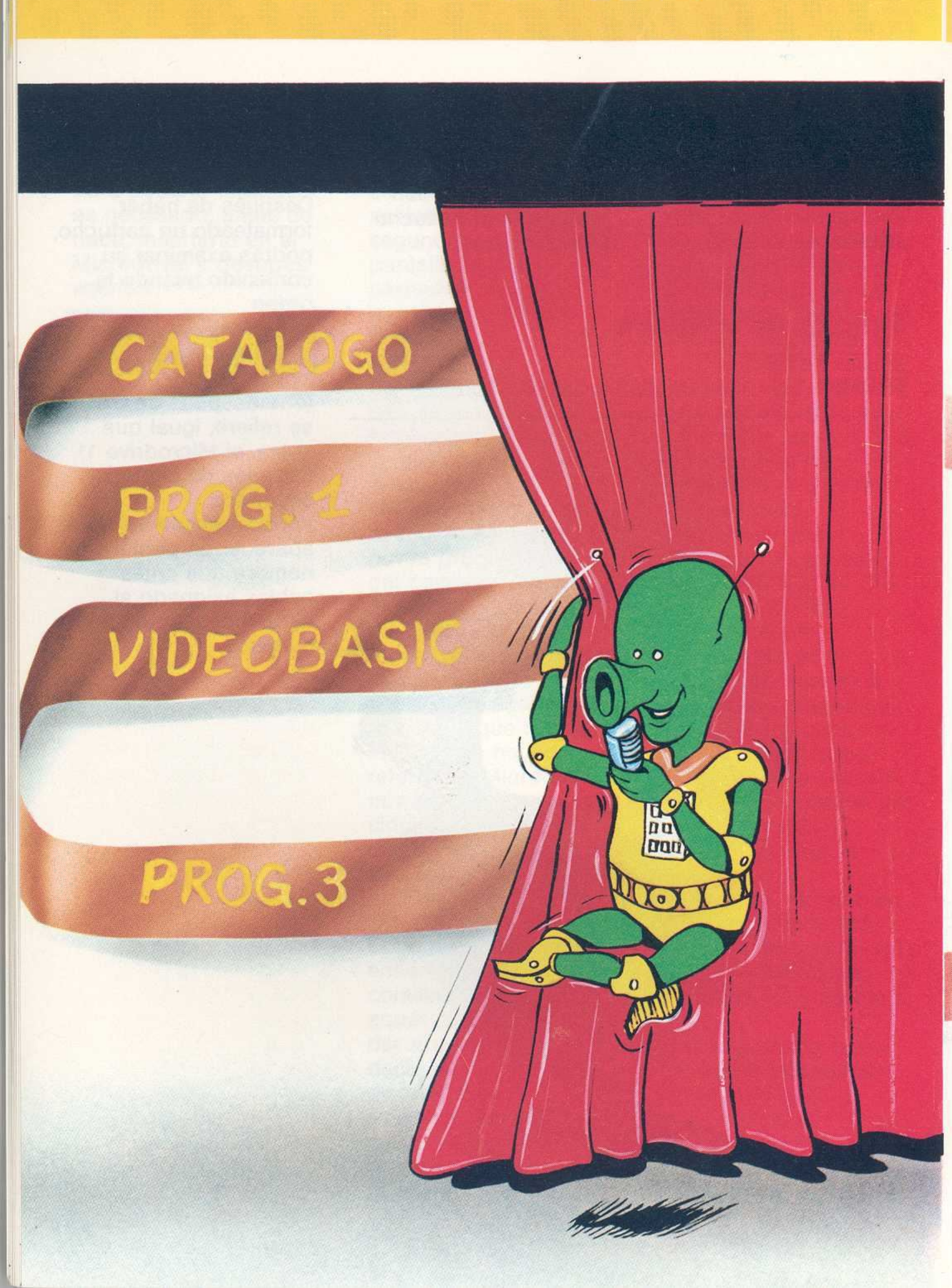
CAT

Después de haber formateado un cartucho, podrás examinar su contenido usando la orden

```
CAT 1
```

(el 1 después de CAT se refiere, igual que antes, al Microdrive 1).
Después de algunos segundos (7 aproximadamente) verás aparecer en pantalla el nombre que antes habías asignado al cartucho, seguido de un número.

LENGUAJE



Este indica cuantos Kilobytes tienes a tu disposición en el cartucho para las sucesivas operaciones de memorización. Según el manual del Microdrive la capacidad de un cartucho está normalmente comprendida entre 85 y 100 Kbytes; en la práctica, suele oscilar en torno a los 90 K. La orden CAT te permite también ver aquéllo que has memorizado en la cinta: por el momento, no habiendo grabado aún nada, indicará solamente el nombre del cartucho y el espacio disponible. Dentro de poco tiempo, cuando hayas aprendido como memorizar los programas, podrás ver el funcionamiento completo de esta instrucción.

Sintaxis de la orden

CAT número Microdrive

SAVE

La gran ventaja que supone el uso del Microdrive se comprueba claramente a través de esta orden. La diferencia de velocidad y de respuesta entre la grabadora y el Microdrive, como en breve podrás comprobar, es verdaderamente notable.
SAVE ejecuta la operación de escritura en el Microdrive prácticamente de la misma manera en que se ejecutaba en la grabadora.
Sin embargo, la

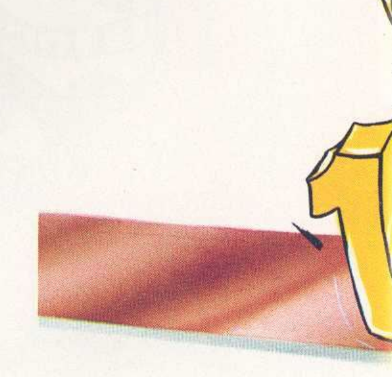
escritura de la orden requiere un poco más de atención.
Mientras que con la grabadora era suficiente teclear, por ejemplo:

```
SAVE "PIPO"
```

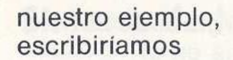
y el programa en memoria era grabado en el casete con el nombre PIPO, ahora es necesario indicar que se desea memorizar el programa en el Microdrive, indicando – igual que ya hemos visto antes– también el número del Microdrive.
La orden necesaria para escribir en el cartucho deberá, por tanto, ser

```
SAVE*"m"; 1; "PIPO"
```

Es importante el asterisco situado detrás de SAVE. La "m" establece que la operación debe ser realizada con el Microdrive; el 1 especifica de qué Microdrive se trata.
El pequeño cansancio necesario para teclear esta instrucción –ligeramente más larga que la relativa a la grabadora– es ampliamente recompensado en el momento de pulsar ENTER.
La grabación de los datos es completamente automática (nunca más START TAPE, THEN PRESS ANY KEY) y, sobre todo, rapidísima.
En el breve período en el que oigas funcionar el motor del lector (eventuales ruidos anómalos indican casi siempre una inserción incorrecta del cartucho), y se encienda la luz roja, tu programa habrá quedado memorizado en el soporte magnético. Una posterior orden CAT te permitirá comprobar que la grabación ha sido efectuada.
Para estar seguro de que todo ha sucedido correctamente es conveniente tener siempre presente la orden VERIFY. En



nuestro ejemplo, escribiríamos


VERIFY*"m"; 1; "PIPO"

El significado y la sintaxis de la instrucción no deberían serte muy difíciles de comprender: es exactamente igual a la primera, excepto en que SAVE se ha convertido en VERIFY.
Si alguna cosa no ha funcionado como debería haberlo hecho, verás aparecer en pantalla el mensaje.

VERIFICATION HAS FAILED

y podrás repetir la operación de grabación, evitando, por un error trivial o un fallo de funcionamiento, la pérdida de todo el programa.

Sintaxis de la orden

SAVE* "m"; número Microdrive; "nombre"

LOAD

Una vez grabado el programa en el cartucho es lógico que se desee tener también la posibilidad de utilizarlo.
Para eso existe LOAD. Hablando siempre de nuestro programa PIPO, la instrucción de lectura del Microdrive deberá ser ahora:

```
LOAD* "m"; 1; "PIPO"
```

También ahora —si no interviene un error de teclado o de inserción del cartucho— tras pocos segundos de funcionamiento, todo estará hecho: el programa será por lo tanto cargado en la memoria de tu Spectrum.
Llegados a este punto no tenemos ya necesidad de alargarnos demasiado: hemos comprobado ya que todo lo que se puede ejecutar con una grabadora, con las modificaciones necesarias, es posible hacerlo también con el Microdrive. La única diferencia es que ahora no podemos dejar de escribir el nombre del programa, como nos permitía hacer el ordenador. Así:

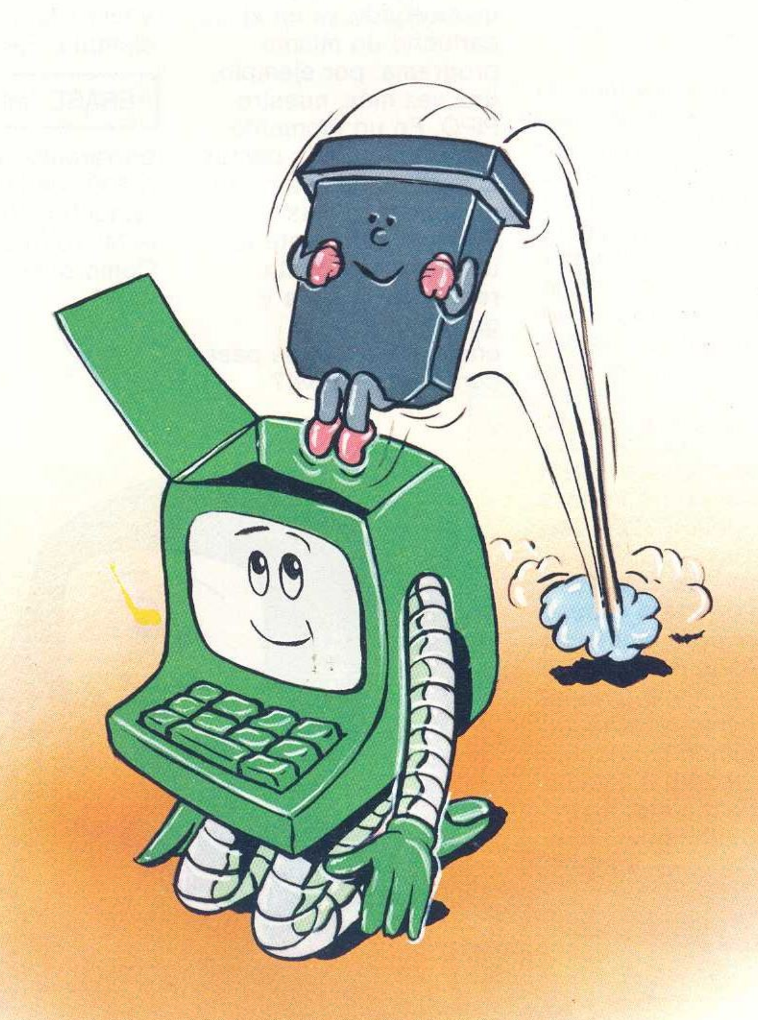
```
LOAD* "m"; 1; " "
```

no cargará el primer programa situado sobre la cinta (entre otras cosas porque al no existir un principio y un final en la cinta, tampoco pueden existir un primero y un último programa), pero provocará el mensaje de error

```
INVALIDE NAME
```

Sintaxis de la orden

LOAD* "m"; número Microdrive; "nombre"



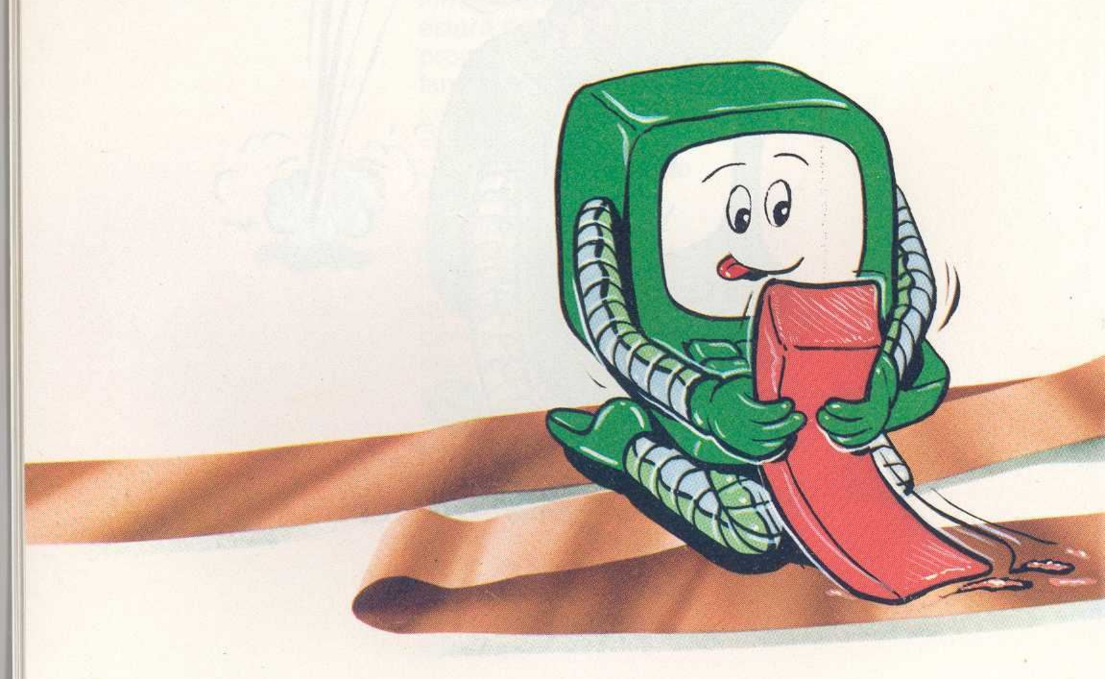
ERASE

Continuando con la comparación entre la grabadora y el Microdrive, veamos ahora para qué sirve ERASE.
Supón que hayamos memorizado ya en el cartucho un mismo programa, por ejemplo, una vez más, nuestro PIPO. En un momento determinado nos damos cuenta de que ya no lo necesitamos más.
Borrarlo del casete es un momento: basta rebobinar la cinta y grabar otra cosa encima. Pero ¿qué pasa con el Microdrive?

Precisamente para esto existe ERASE. Esta orden elimina del cartucho el programa, o, más genéricamente, los archivos, que hayamos especificado. Veámoslo con un ejemplo. Ejecutando:

```
ERASE "m"; 1; "PIPO"
```

borraremos el archivo "PIPO" de la cinta del cartucho introducido en el Microdrive 1.
Como puedes



comprobar, a diferencia de las otras órdenes del Microdrive, ERASE no necesita el asterisco. Naturalmente, para que pueda ser borrado, el archivo debe estar presente en el cartucho; las tentativas de borrar archivos inexistentes acarrearán necesariamente el mensaje:

FILE NOT FOUND

(archivo no encontrado)

La operación de borrado debe hacerse siempre antes de la puesta al día de cualquier programa. Por ejemplo, si quisieras memorizar una nueva versión de PIPO sin haber borrado antes la anterior, obtendrás un error del tipo:

WRITING TO A "READ" FILE

(estás tratando de escribir en un archivo de "LECTURA").

Así, todos las veces que desees grabar un programa cuyo nombre existe ya en el cartucho tendrás dos posibilidades: o cambiar el nombre del archivo a grabar, o borrar antes el viejo archivo del Microdrive.

Sintaxis de la orden

ERASE "m"; número Microdrive; "nombre"

Uso de los archivos secuenciales

En esta parte de la lección profundizaremos en el tema de los archivos que hemos comenzado antes.
En primer lugar, antes de poder acceder a un archivo es necesario abrirlo.

Abrir un archivo significa ordenar a nuestro Spectrum que busque las informaciones referentes a dicho archivo: si está en la cinta, y, si es así, dónde se encuentra exactamente.
La orden OPEN hace que el Microdrive ejecute todas estas operaciones, reservando además una cierta área en la memoria del ordenador que será usada como buffer (llamada también memoria tampón). El buffer no es más que un depósito de informaciones cuya misión es permitir al Microdrive acceder sólo de vez en cuando a los cartuchos, y no cada vez que el programa tiene un solo dato que leer o escribir. En otras palabras, el acceso a la cinta tiene lugar a través de bloques de informaciones y no por datos independientes, con el consiguiente gran ahorro de tiempo.
Un ejemplo de la orden OPEN para la apertura de un archivo secuencial es el siguiente:

```
OPEN # 4; "m"; 1; "PRUEBA"
```

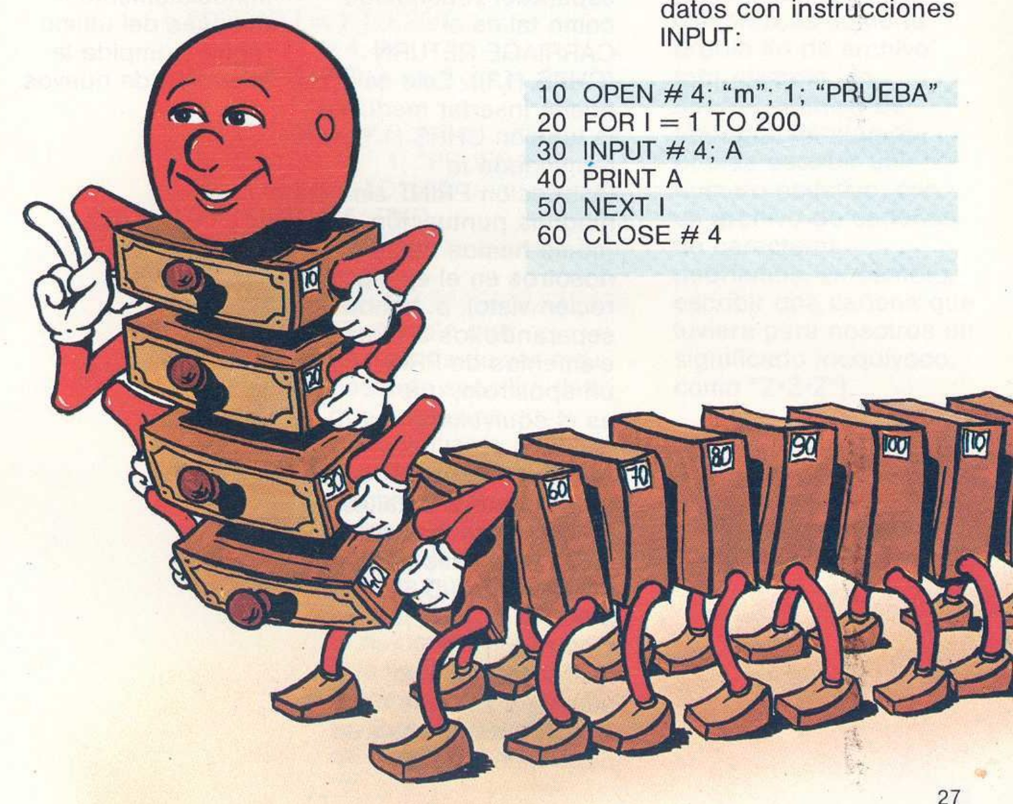
Al llegar a este punto, todas las instrucciones PRINT # que se refieran al canal recién abierto provocarán la memorización de datos en el cartucho. Este breve programa ilustra el funcionamiento de todo lo que acabamos de decir:

```
10 OPEN # 4; "m"; 1;
   "PRUEBA"
20 FOR I = 1 TO 200
30 LET A = RND
40 PRINT A
50 PRINT # 4; A
60 NEXT I
70 CLOSE # 4
```

Tras la apertura del archivo serán enviados 200 números al azar respectivamente a la pantalla y al Microdrive; al final del ciclo el archivo será cerrado y los números habrán sido memorizados en la cinta magnética.
Inmediatamente después del último número, siguiendo a CLOSE, será

memorizado también un carácter de EOF (End Of File, es decir, fin del archivo) necesario para que el carácter pueda gestionar este archivo en el futuro, cuando quieras recuperar lo que has escrito.
La importancia de CLOSE, y no sólo por el carácter EOF, es relevante incluso viendo trabajar a este programa.
En efecto, podrás comprobar que la luz roja del microdrive no permanecerá siempre encendida: las informaciones serán grabadas solamente cuando el buffer haya sido llenado dato tras dato en el curso de la ejecución. Si no hubieramos enviado la orden CLOSE, los datos que permanezcan contenidos en el buffer al terminar el ciclo no habrían llegado nunca al cartucho, ya que el ordenador estaría esperando nuevas "llegadas" y el archivo habría quedado incompleto.
Por lo tanto, siempre es necesario acordarse de cerrar los archivos.
Veamos a continuación qué es lo que hay que hacer para recuperar los números que acabamos de memorizar en el microdrive. En lugar de enviar PRINT ahora deberemos pedir los datos con instrucciones INPUT:

```
10 OPEN # 4; "m"; 1; "PRUEBA"
20 FOR I = 1 TO 200
30 INPUT # 4; A
40 PRINT A
50 NEXT I
60 CLOSE # 4
```



El funcionamiento del programa será idéntico al visto con anterioridad, excepto que el ordenador en lugar de escribir, leerá en el microdrive.
Revisten una especial importancia en el momento de la escritura de los archivos los caracteres de separación entre los distintos campos. Para poder leer los datos mediante instrucciones INPUT es necesario que los campos sean grabados figurando a continuación de cada uno de ellos un carácter separador válido, es decir, que pueda ser reconocido como tal por el microdrive.
El principal carácter separador reconocido como tal es el CARRIAGE RETURN (CHR$ (13)). Este se puede insertar mediante la función CHR$ (13), o terminando la instrucción PRINT sin ninguna puntuación (como hemos hecho nosotros en el ejemplo recién visto), o, también separando los distintos elementos de PRINT con un apóstrofe, que es el equivalente de ENTER.
Sin duda, lo mejor, y sin entrar en más detalles, es no realizar nunca PRINT múltiples (del tipo PRINT # 4, A, B, C o INPUT # 4, A, B, C), sino recurrir a instrucciones más sencillas. Aunque así los programas sean un poco más largos, la facilidad de uso de esta solución nos permite dominar siempre la situación, por lo menos hasta que se haya adquirido la suficiente soltura en el manejo de archivos.
Otro aspecto a señalar es que la puesta al día de un archivo secuencial sólo puede conseguirse mediante una nueva reescritura completa de éste, puesto que la presencia de EOF, situado inmediatamente después del último registro, impide la inserción de nuevos

elementos al final de la lista.
Así pues, lo que hay que hacer es lo siguiente:
– abrir para lectura el archivo a poner al día;
– abrir para escritura un nuevo archivo;
– copiar ordenadamente los registros del viejo en el nuevo archivo aportando cada vez las modificaciones o añadidos que se consideren necesarios;
– cerrar al final los dos archivos y borrar el viejo.
Veamos como ejemplo lo que deberiamos de hacer para añadir otros 100 números al azar al archivo que habíamos creado anteriormente:

```
 10 OPEN # 4; "m"; 1; "PRUEBA"
 20 OPEN # 5; "m"; 1; "PRUEBA 2"
 30 FOR I = 1 TO 200
 40 INPUT # 4; A
 50 PRINT # 5; A
 60 NEXT I
 70 CLOSE# 4
 80 ERASE "m"; 1; "PRUEBA"
 90 FOR I = 1 TO 100
100 PRINT # 5; RND
110 NEXT I
120 CLOSE # 5
```

El ejemplo que acabamos de ver era bastante sencillo, dado que sabíamos perfectamente cuantos números teníamos que transferir del viejo al nuevo archivo antes de pasar a la extracción de nuevos números al azar.
La mayoría de las veces el número de lecturas que hay que realizar antes de llegar al EOF es completamente desconocido y ten en cuenta además que intentar leer más allá del último registro provoca un mensaje de error. Por lo tanto, no es posible leer los registros "al azar".
Un método casi infalible para solventar el problema es el siguiente: será necesario modificar el programa para escribir en el archivo, como último registro uno o más caracteres especiales, o fácilmente reconocibles, que tú establezcas como tu propio fin de archivo (por ejemplo, en nuestro archivo de números habríamos podido escribir un número negativo; con un archivo de cadenas de caracteres podríamos, en cambio, escribir una cadena que tuviera para nosotros un significado inequívoco, como "Z*Z*Z").

De esta manera, en la fase de lectura será suficiente con insertar una instrucción que compruebe el fin de archivo, expresión que se verificará en el momento en que el carácter especial sea detectado por el programa.
Veamos un ejemplo al respecto: el programa siguiente escribirá en el cartucho un archivo con tantas cadenas como las que introduzcamos por el teclado.
Cuando decidas acabar te será suficiente con pulsar ENTER.
Antes de cerrar el archivo el ordenador escribirá automáticamente como último registro nuestro código especial "Z*Z*Z".

```
10 OPEN # 4; "m"; 1; "PRUEBA3"
20 INPUT "ESCRIBE LA CADENA (O SOLO ENTER PARA TERMINAR)"; A$
30 IF A$ = " " THEN GOTO 60
40 PRINT # 4; A$
50 GOTO 20
60 PRINT # 4; "Z*Z*Z"
70 CLOSE # 4
```

Ahora podrás recuperar y leer las cadenas que acabas de teclear con este segundo programa.

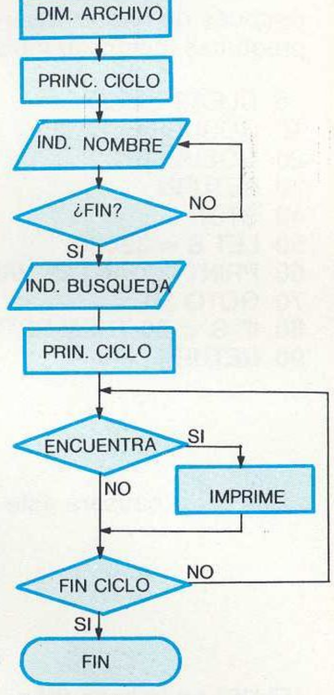
```
10 OPEN # 4; "m"; 1; "PRUEBA3"
20 INPUT # 4; A$
30 IF A$ = "Z*Z*Z" THEN CLOSE # 4: STOP
40 PRINT A$
50 GOTO 20
```

Ahora ya no es necesario conocer el número de elementos existentes en el archivo, nuestro "Z*Z*Z" advierte a la unidad central de

que no debe intentar leer más allá puesto que ya no existen más registros. De esta manera nos resguarda de un posible error.

Búsqueda sobre archivos

El programa analizado a continuación hace referencia a una aplicación típicamente informática: la memorización y búsqueda de informaciones.
En la fase preliminar memorizaremos 9 informaciones cualesquiera (nombres, palabras, números).
A continuación localizaremos aquéllas que deseemos en base a un argumento de búsqueda que también puede ser sólo parcial.




```
10 DIM F$ (9,10)
20 FOR I = 1 TO 9
30 INPUT "NOMBRE"; F$ (I)
40 NEXT I
50 INPUT "BUSCA"; C$
60 FOR I=1 TO 9
70 IF F$ (I, TO LEN C$) = C$ THEN PRINT F$ (I)
80 NEXT I
```



El error siempre acecha: espera encantado cualquier distracción que puedas cometer, aunque sea mínima. Nunca le des la oportunidad de reirse de tí. Apréndete la sintaxis de las instrucciones. Haz los programas de forma modular, fácilmente legibles y muy documentados. Y además, pon el máximo cuidado a la hora de teclear, hasta una simple coma en un lugar equivocado puede comprometer el correcto funcionamiento de un programa.

EJERCICIOS

Después de haber observado el listado con toda atención contesta a las preguntas, comprobando personalmente tus respuestas mediante el teclado.

```
 5 CLEAR 24500
10 GOSUB 50
20 GOSUB 10
30 RETURN
40 STOP
50 LET S = 32400
60 PRINT "CUANTAS VARIABLES..."
70 GOTO 30
80 IF S < 56 THEN RETURN
90 RETURN
```

¿Qué error causará este programa?

¿El "IF" en la línea 80 podrá ser alguna vez verdadero?

¿El programa se parará con "STOP"... o no?

¿Dado el CLEAR en la línea 5 cuántos GOSUB sin RETURN podrá soportar la pila (STACK) de los GOSUB?